

Narration	Time
Ruby میں for اور each loops کے اسپوکن ٹیوٹوریل میں خوش آمدید۔	00:01
اس ٹیوٹوریل میں ہم سیکھیں گے	00:05
لفظ loop کا مطلب	00:07
Ruby میں مختلف قسم کے لوپس	00:08
for لوپ کا استعمال اور	00:11
each لوپنگ کنسٹرکٹ کا استعمال	00:12
یہاں ہم استعمال کر رہے ہیں	00:14
Ubuntu ورژن 12.04	00:16
اور Ruby 1.9.3	00:19
اس ٹیوٹوریل کو سمجھنے کے لئے، آپ کے پاس انٹرنیٹ کنکشن ہونا ضروری ہے۔	00:22
آپ کو Linux کمانڈس، ٹرمینل اور ٹیکسٹ-ایڈیٹر کا علم بھی ہونا چاہئے۔	00:25
اگر نہیں، تو متعلقہ ٹیوٹوریل کے لئے، ہماری ویب سائٹ ملاحظہ کریں۔	00:30
اب میں سمجھاتا ہوں کہ Ruby میں loop کسے کہا جاتا ہے۔	00:34
loop، کمانڈ یا کمانڈس کا سیٹ ہے جو مقررہ بار بار ایکریوٹ ہوتا ہے۔	00:38
Ruby مندرجہ ذیل اہم اسٹیٹمنٹس رکھتا ہے	00:44
for	00:47
each	00:48
while	00:49
until	00:49
اس ٹیوٹوریل میں ہم سیکھیں گے کہ for اور each لوپنگ کنسٹرکٹ کو کس طرح استعمال کریں۔	00:50
شروع کرنے سے پہلے، براہ مہربانی اپنی ہوم ڈائریکٹری میں مندرجہ ذیل فولڈرز بنائیں۔	00:55
ttr, ruby hyphen tutorial, looping hyphen statements.	01:02
تمام فولڈرز بنائیں جیسا دکھایا گیا ہے۔	01:07

01:11	اب ہم نے ضروری فولڈر بنائے ہیں۔
01:13	آگے بڑھتے ہیں۔
01:15	Ruby میں for لوپ کا سنٹیکس ہے۔
01:19	'for variable in a collection of objects
01:22	ruby code
01:23	end
01:25	اس کو ایک مثال کے ذریعے سیکھتے ہیں۔
01:28	gedit میں ایک نئی فائل بناتے ہیں جیسے ابتدائی Ruby ٹیوٹوریلز میں دکھایا گیا ہے
01:32	اور اس کو for hyphen loop dot rb نام دیں۔
01:36	میرے پاس پہلے سے ہی for لوپ کی ایک فعال مثال ہے۔
01:39	اس مثال کو پورا کرنے کے دوران آپ کو ڈٹا ٹپ کر سکتے ہیں۔
01:44	اس مثال میں میں نے ایک for لوپ ڈکلیئر کیا ہے۔
01:47	ہمارے پاس 1 سے 20 تک نمبرز کا سیٹ ہے۔
01:50	ہم، for لوپ میں ایک ویریبل i ڈکلیئر کرتے ہیں۔
01:55	1 سے 20 تک نمبرز کے سیٹ میں i کو پہلے ایلمینٹ سے انشلائز یعنی شروع کیا جاتا ہے۔
02:00	for لوپ کی ڈکلیئریشن کی وجہ سے یہ کوڈ 1 سے 20 تک ہر نمبر پر بار بار جاتا ہے۔
02:07	for لوپ میں ڈکلیئر puts میٹھڈ آؤٹ پٹ کے لئے ذمہ دار ہے۔
02:14	اب ٹرمینل کھولیں اور ٹائپ کریں۔
02:17	ruby space for hyphen loop dot rb اور آؤٹ پٹ دیکھیں۔
02:22	آؤٹ پٹ 1 سے 20 کے نمبرز کا array ہوگا
02:26	اس مثال میں، ہم نے inclusive range کے لئے for لوپ ڈکلیئر کیا۔
02:31	اس میں 1 سے 20 تک کے نمبرز داخل ہیں۔
02:35	اب، ہم non-inclusive range کے لئے for لوپ کا نفاذ دیکھیں گے۔
02:41	کوڈ کا اگلا حصہ ٹائپ کرنا جاری رکھیں۔

02:44	non-inclusive range کا مطلب ہے کہ یہ آنجیکٹس کے گروپ میں آخری ایلمینٹ داخل نہیں کرے گا۔
02:49	یہاں for لوپ، 1 سے 20 تک کی non-inclusive range کے لئے لاگو ہوا ہے۔
02:55	آپ جلد ہی دیکھیں گے کہ نمبر 20 آؤٹ پٹ میں پرنٹ نہیں ہوگا۔
02:59	آخری نمبر داخل نہیں ہوتا، جب آپ کے پاس نمبرز کے درمیان میں تین نقطے ہوں
03:04	اب ٹرمینل کھولیں اور ٹائپ کریں. ruby space for hyphen loop dot rb
03:13	اور آؤٹ پٹ دیکھیں۔
03:17	آؤٹ پٹ اب بھی نمبرز کا array ہے، لیکن نمبر 20 داخل نہیں ہوگا۔
03:22	اب، آپ اپنا for لوپ لکھنے کے لئے کافی حد تک قابل ہونے چاہئے۔
03:27	Ruby میں each لوپ کا سٹیکس مندرجہ ذیل ہے:
03:31	a collection of objects' dot each do item
03:36	Ruby کوڈ end
03:38	اس کو ایک مثال سے سمجھاتے ہیں۔
03:42	gedit میں ایک نئی فائل بنائیں، جیسے کہ Ruby کے ابتدائی ٹیوٹوریلز میں دکھایا گیا ہے۔
03:46	اور اس کو each hyphen loop dot rb نام دیں
03:50	میرے پاس each لوپ کی ایک فعال مثال ہے۔
03:53	اس مثال کو سمجھتے وقت آپ کوڈ بھی ٹائپ کر سکتے ہیں۔
03:58	میں نے اس مثال میں ایک each لوپ ڈکلیئر کیا ہے۔
04:03	ہمارے پاس 1 سے 20 تک نمبرز کا سیٹ ہے۔
04:06	each لوپ میں، ہم ایک ویریبل 'i' ڈکلیئر کرتے ہیں۔
04:11	1 سے 20 تک نمبرز کے سیٹ میں i کو پہلے ایلمینٹ سے انشلائز کیا جاتا ہے۔
04:17	each لوپ کی ڈکلیئریشن کی وجہ سے یہ کوڈ 1 سے 20 تک ہر نمبر پر بار بار جاتا ہے۔
04:23	each میں puts میٹھڈ ڈکلیئر کیا ہے، جو آؤٹ پٹ کے لئے ہے۔
04:30	اب ٹرمینل کھولیں اور ٹائپ کریں. ruby space each hyphen loop dot rb
04:39	اور آؤٹ پٹ دیکھیں۔

04:43	آؤٹ پٹ 1 سے 20 کے نمبرس کا array ہے
04:46	مندرجہ بالا مثال میں ہم نے inclusive range کے لئے ایک each لوپ ڈکلیئر کیا ہے۔
04:51	اس میں 1 سے 20 تک سارے نمبرس داخل ہیں۔
04:54	اب، ہم non-inclusive range کے لئے each لوپ کے نفاذ دیکھیں گے۔
05:00	کوڈ کے اس حصے کو ٹائپ کرنا جاری رکھیں۔
05:04	Non-inclusive کا مطلب، یہ آہنجیکٹس کے مجموعے میں آخری ایلیمنٹ داخل نہیں کرے گا۔
05:10	یہاں each لوپ، 1 سے 20 نمبرس تک کی non-inclusive range کے لئے لاگو ہوا ہے۔
05:16	جلد ہی آپ دیکھیں گے کہ نمبر 20 آؤٹ پٹ میں پرنٹ نہیں ہوگا۔
05:20	جب نمبرس کے درمیان 3 داٹس یعنی نقطے ہوں، تو آخری نمبر داخل نہیں ہوتا
05:25	اب ٹرمینل کھولیں اور ٹائپ کریں ruby space each hyphen loop dot rb
05:34	اور آؤٹ پٹ دیکھیں۔
05:39	اب، آپ مناسب طریقے سے اپنا each لوپ لکھنے کے قابل ہونے چاہئیں۔
05:44	ہم کس طرح کا تعین کریں کہ کون سا لوپنگ construct انتخاب کرنا ہے؟
05:48	"for" loop construct کو دوبارہ کال کرتے ہیں۔
05:53	پہلی مثال میں، for استعمال کر کے ہم نے 1 سے 20 تک نمبرس کے سیٹ کو دہرایا۔
05:59	اپنے ٹرمینل پر ruby space for hyphen loop dot rb کو ایکریوٹ کریں، اور آؤٹ پٹ دیکھیں۔
06:08	اب gedit میں کوڈ دیکھتے ہیں۔
06:11	جب آپ for لوپ کا استعمال کرتے ہیں، تو Ruby پیچھے سے each طریقہ کو کال کرتا ہے۔
06:16	اور each یا for کو کال کرنا یکساں آؤٹ پٹ دیتا ہے
06:21	چونکہ for بھی، each کو کال کرتا ہے لہذا اس کے بجائے each لوپ کا استعمال کرنا بہتر ہے۔
06:28	یہ ہمیں اس سبق کے آخر میں لاتا ہے۔
06:30	اس کا خلاصہ کرتے ہیں۔
06:32	اس ٹیوٹوریل میں ہم نے سیکھا
06:33	for لوپ کا استعمال

06:35	each کنسٹرکٹ کو استعمال
06:38	for کے بجائے each استعمال کرنے کی وجہ
06:41	مندرجہ بالا لوپنگ کنسٹرکٹ استعمال کرنے کی مثالیں
06:45	ایک مشق
06:47	مناسب لوپ کنسٹرکٹ استعمال کرتے ہوئے ایک ruby پروگرام لکھیں۔
06:50	1 سے 20 تک نمبرز کے سیٹ سے even نمبرز کا آؤٹ پٹ بنائیں۔
06:56	مندرجہ ذیل لنک پر دستیاب ویڈیو دیکھیں۔
07:00	یہ اسپوکن ٹیوٹوریل پروجیکٹ کا خلاصہ کرتا ہے۔
07:03	اچھی بینڈ ویڈیو نہ ہونے پر آپ اسے ڈاؤن لوڈ کر کے دیکھ سکتے ہیں۔
07:08	اسپوکن ٹیوٹوریل پروجیکٹ ٹیم:
07:11	اسپوکن ٹیوٹوریلز کا استعمال کرتے ہوئے ورکشاپ چلاتی ہے۔
07:14	اور آن لائن ٹیسٹ پاس کرنے والوں کو ٹیفلیٹ دیتی ہے۔
07:18	مزید معلومات کے لئے، contact@spoken-tutorial.org پر لکھیں۔
07:23	اسپوکن ٹیوٹوریل پروجیکٹ ٹاک ٹوائے ٹیچر پراجیکٹ کا حصہ ہے۔
07:26	یہ بھارتی حکومت کے ایم ایچ آر ڈی کے آئی سی ٹی کے ذریعے قومی خواندگی مشن کی طرف سے حمایت شدہ ہے۔
07:33	اس مشن پر مزید معلومات spoken hyphen tutorial dot org slash NMEICT hyphen Intro دستیاب ہیں۔
07:41	اس اسکرپٹ کا ترجمہ اور صدا بندی میں نے یعنی وجاحت احمد نے کی ہے۔ شامل ہونے کیلئے آپ کا شکریہ۔